A Framework for Efficient and Composable Oblivious Transfer

Chris Peikert[1] Vinod Vaikuntanathan[2] Brent Waters[1]

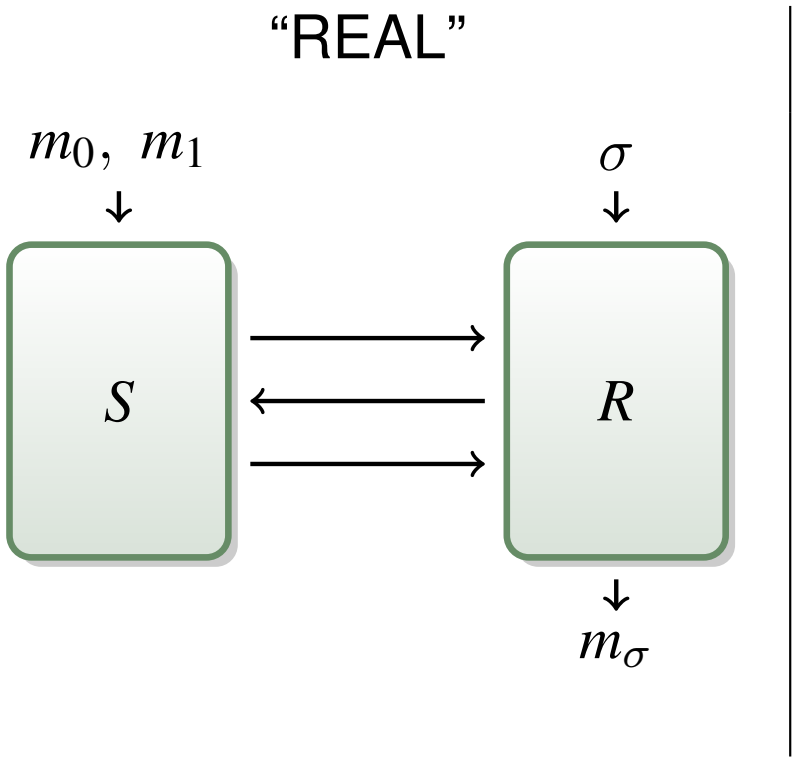
[1]SRI International

[2]MIT

CRYPTO 2008

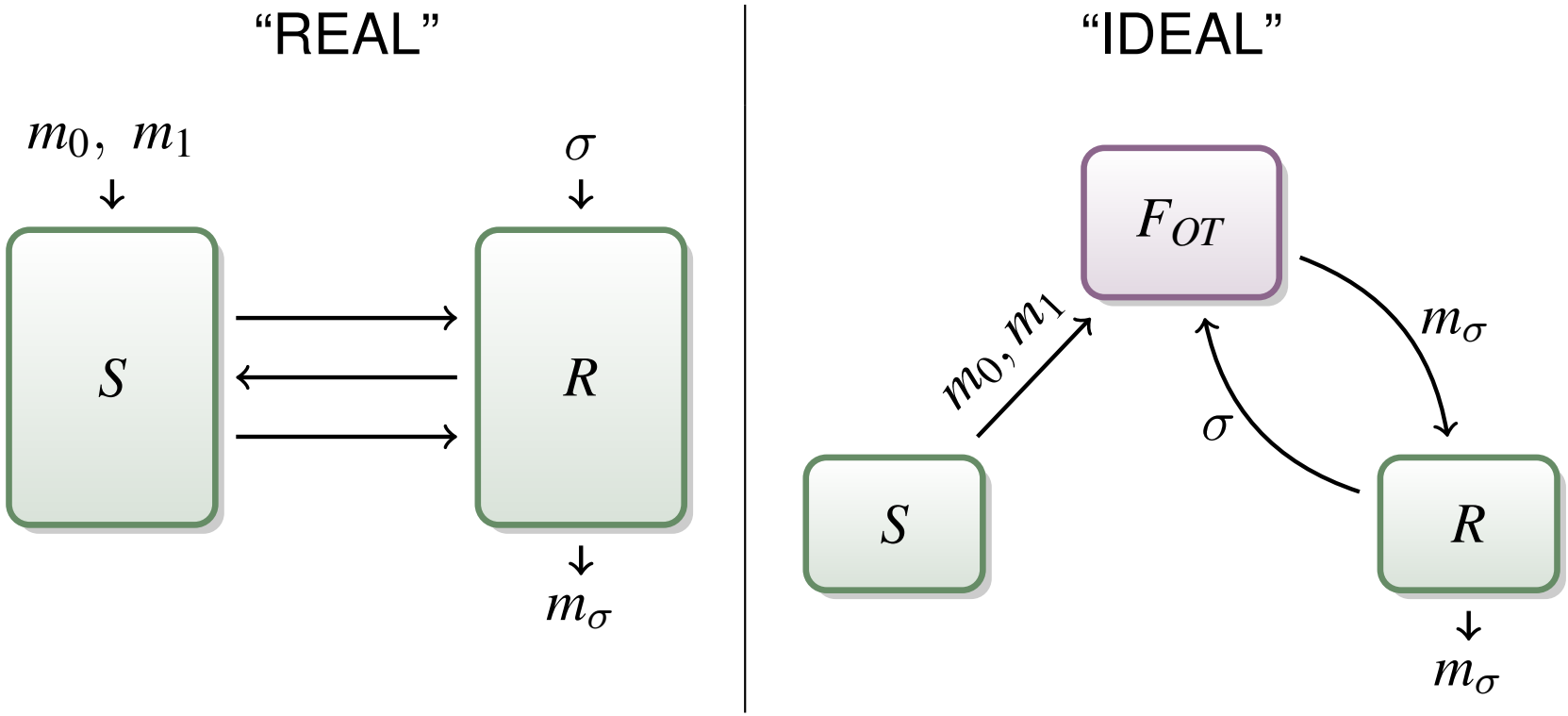
Oblivious Transfer

[R81,EGL85,BCR86,C87,K88,CK88,C89,CS91,CGT95,DCP95,FMR96,BC97,. . .]



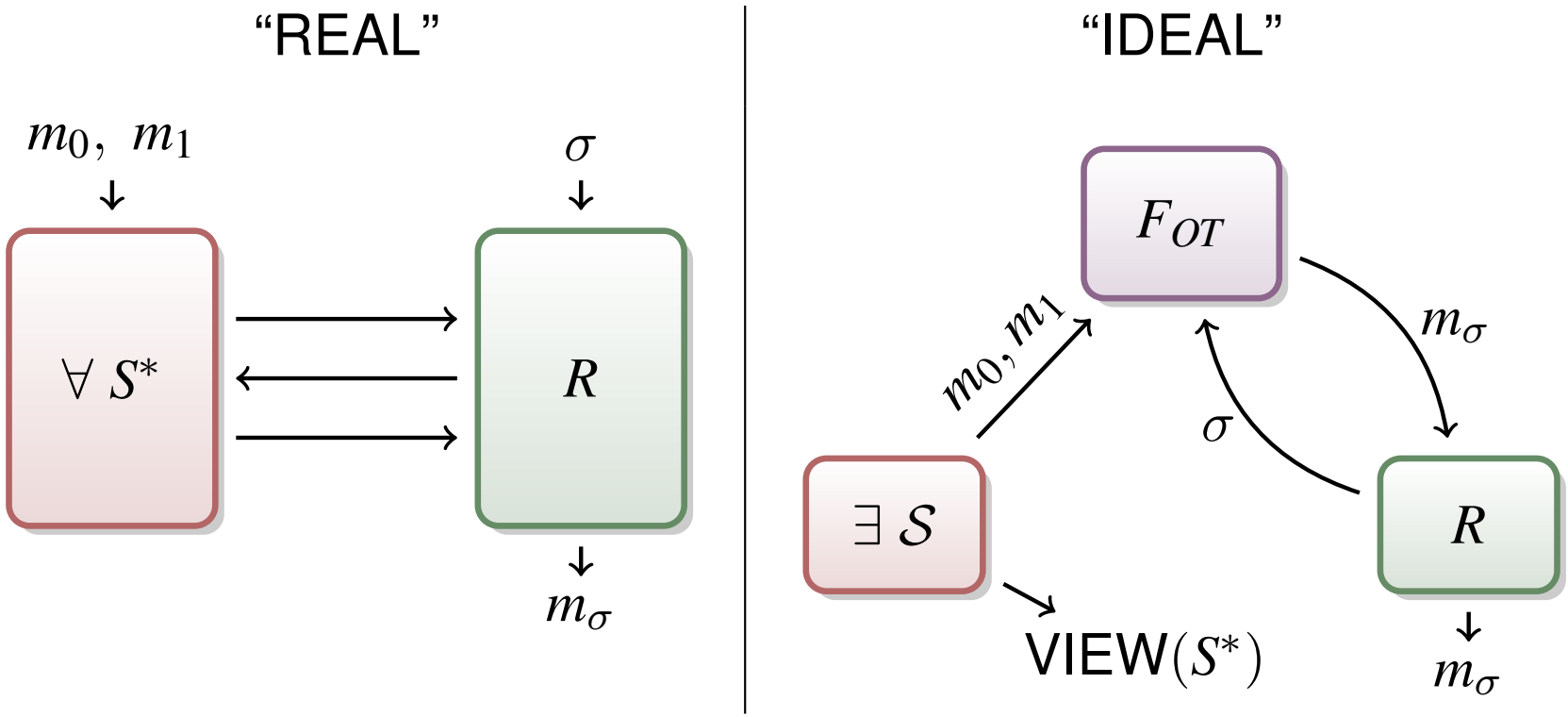
Oblivious Transfer

[R81,EGL85,BCR86,C87,K88,CK88,C89,CS91,CGT95,DCP95,FMR96,BC97,. . .]



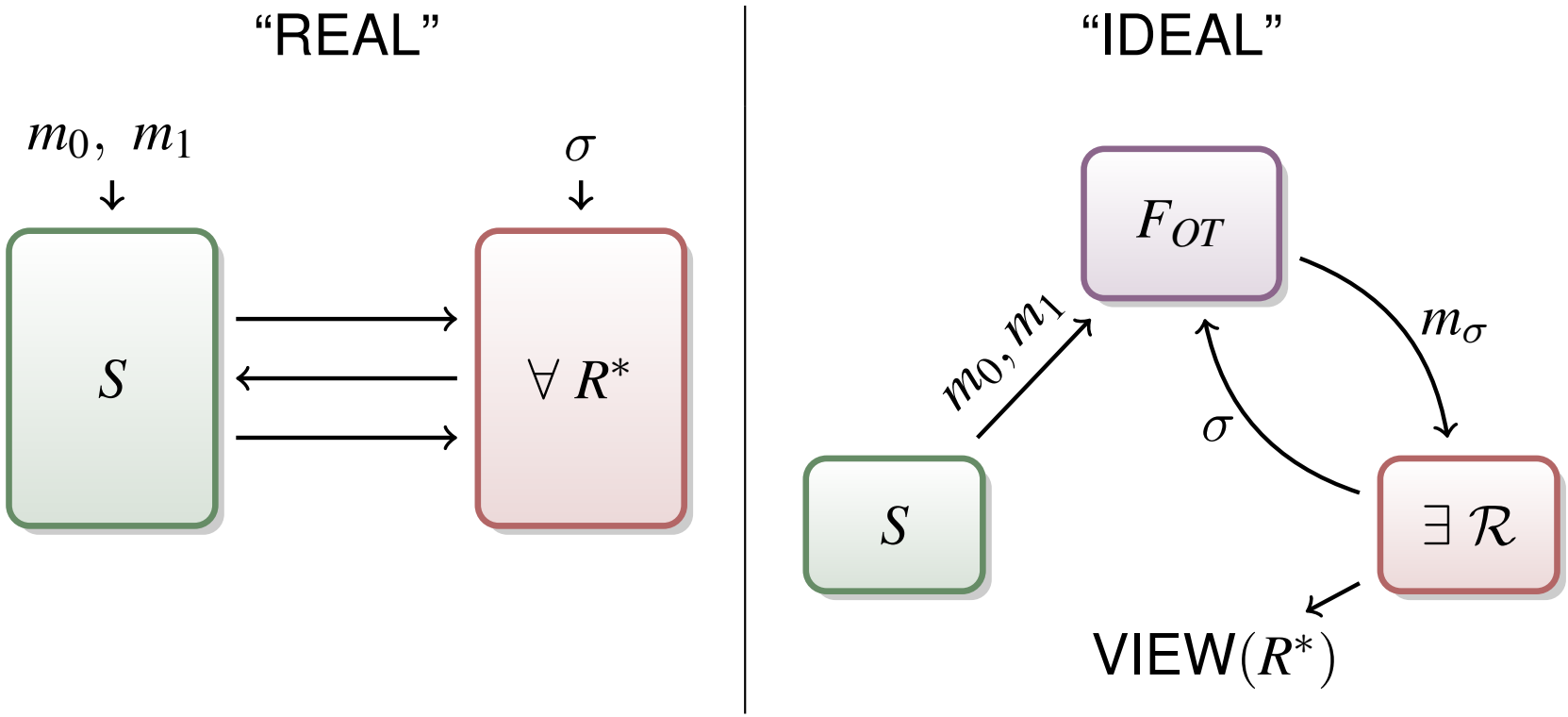
Oblivious Transfer

[R81,EGL85,BCR86,C87,K88,CK88,C89,CS91,CGT95,DCP95,FMR96,BC97,...]



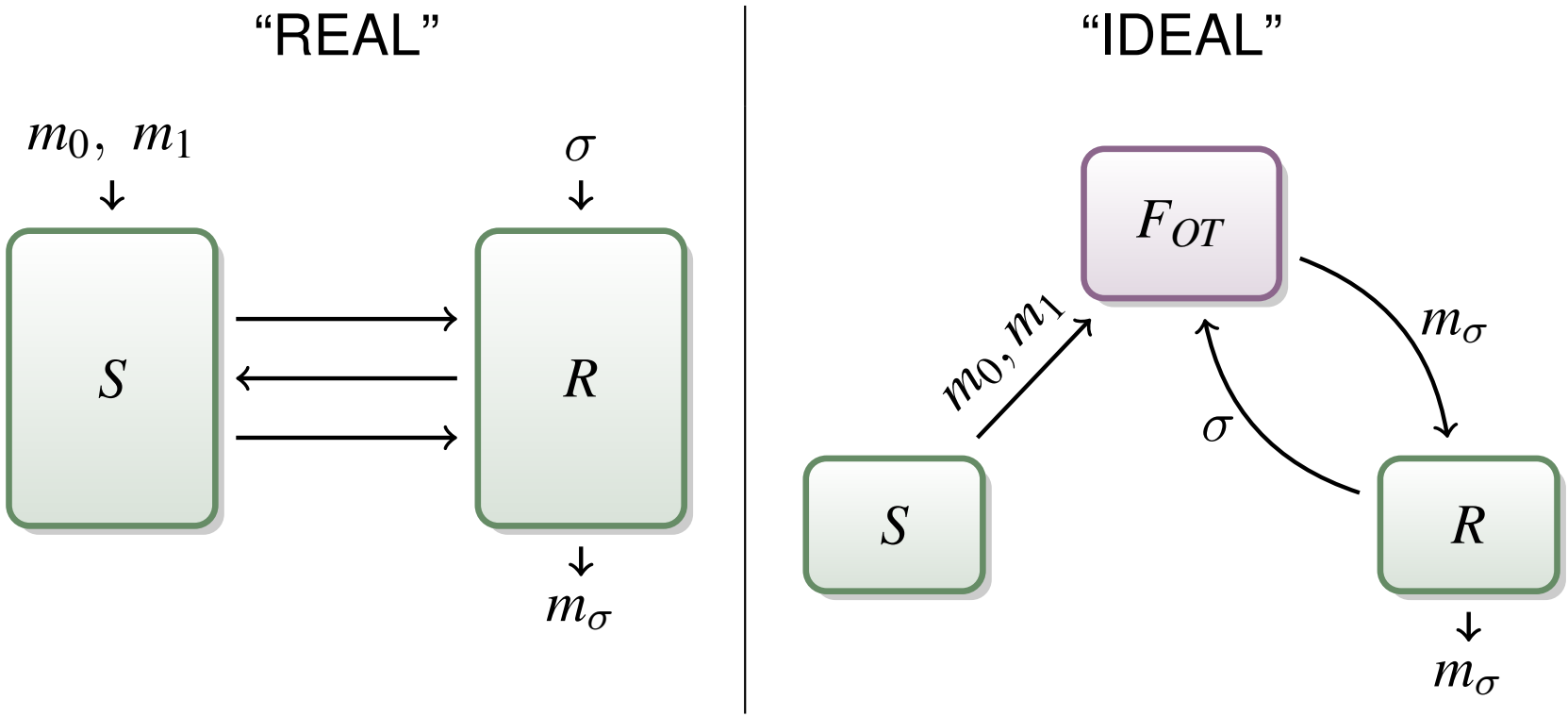
Oblivious Transfer

[R81,EGL85,BCR86,C87,K88,CK88,C89,CS91,CGT95,DCP95,FMR96,BC97,...]



Oblivious Transfer

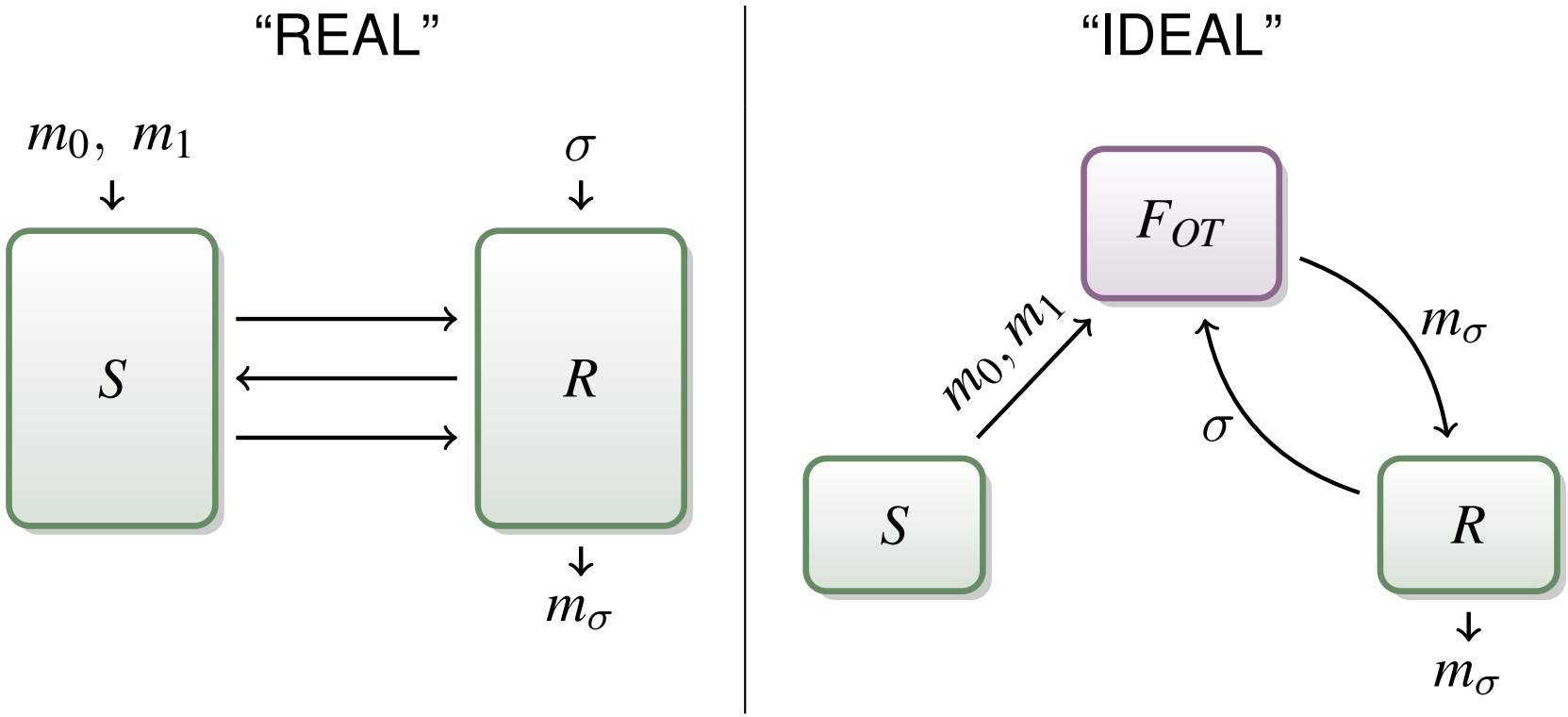
[R81,EGL85,BCR86,C87,K88,CK88,C89,CS91,CGT95,DCP95,FMR96,BC97,...]



- 'Complete' for secure computation [Yao82,GMW87,Kil88]

Oblivious Transfer

[R81,EGL85,BCR86,C87,K88,CK88,C89,CS91,CGT95,DCP95,FMR96,BC97,...]



- 'Complete' for secure computation [Yao82,GMW87,Kil88]
- Feasible: (enhanced) TDPs + zero knowledge [EGL85,GMW86]

Prior Efficient Protocols

- Two messages: [NP01,AIR01,Tau05,CS02]

Prior Efficient Protocols

- Two messages: [NP01,AIR01,Tau05,CS02] – *'half simulation'*

Prior Efficient Protocols

- Two messages: [NP01,AIR01,Tau05,CS02] – *'half simulation'*
- Full simulation: [Lin08]

Prior Efficient Protocols

- Two messages: [NP01,AIR01,Tau05,CS02] – *'half simulation'*
- Full simulation: [Lin08] – *blowup, extra rounds*

Prior Efficient Protocols

- Two messages: [NP01,AIR01,Tau05,CS02] – *'half simulation'*
- Full simulation: [Lin08] – *blowup, extra rounds*
- 'Adaptive selection:' [CNS07,GH07] – bilinear, many rounds

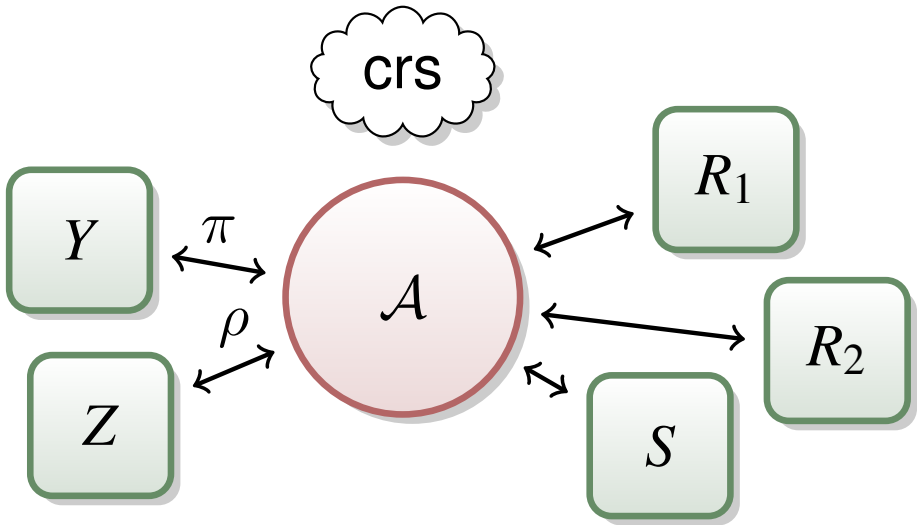
Prior Efficient Protocols

- Two messages: [NP01,AIR01,Tau05,CS02] – *'half simulation'*
- Full simulation: [Lin08] – *blowup, extra rounds*
- 'Adaptive selection:' [CNS07,GH07] – bilinear, many rounds

Will it *COMPOSE* ?

UC framework [Can01]

Requires *setup* [CF01]



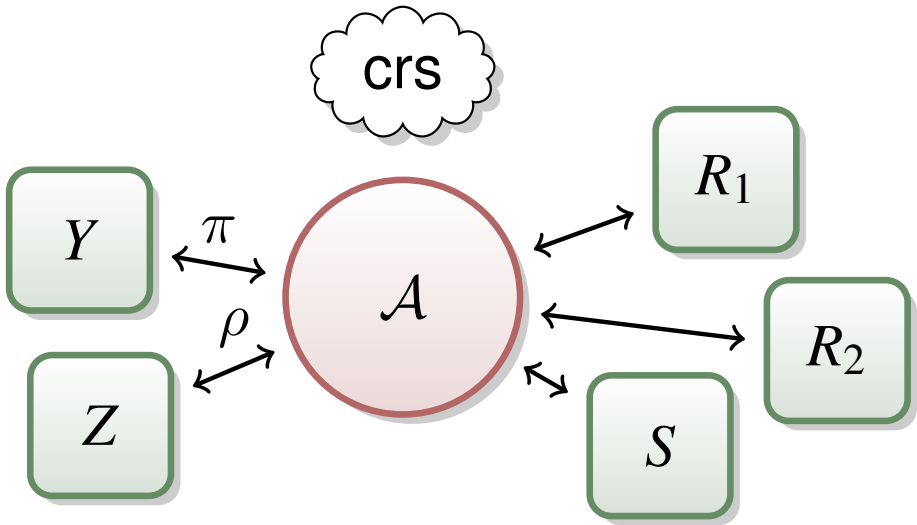
Prior Efficient Protocols

- Two messages: [NP01,AIR01,Tau05,CS02] – *'half simulation'*
- Full simulation: [Lin08] – *blowup, extra rounds*
- 'Adaptive selection:' [CNS07,GH07] – bilinear, many rounds

Will it *COMPOSE* ?

UC framework [Can01]

Requires *setup* [CF01]



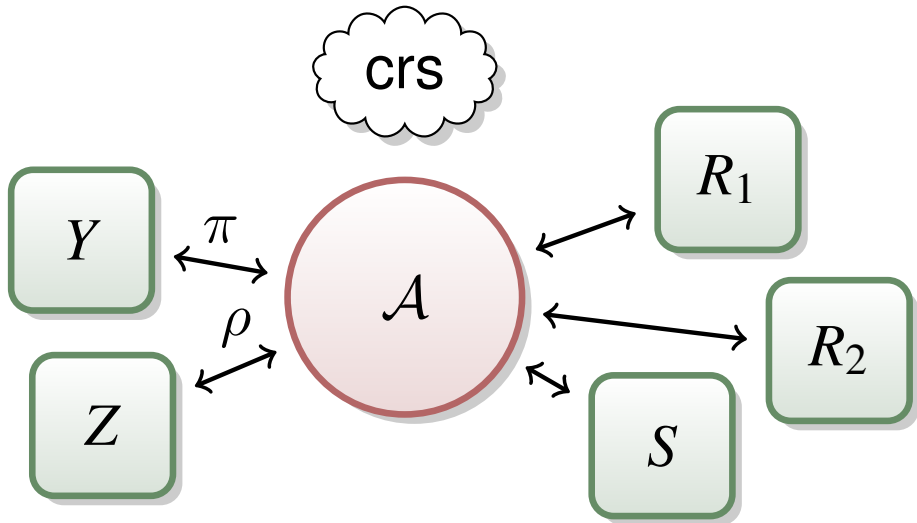
Prior Efficient Protocols

- Two messages: [NP01,AIR01,Tau05,CS02] – *'half simulation'*
- Full simulation: [Lin08] – *blowup, extra rounds*
- 'Adaptive selection:' [CNS07,GH07] – bilinear, many rounds

Will it *COMPOSE* ?

UC framework [Can01]

Requires *setup* [CF01]



COMPOSABILITY aids *EFFICIENCY*

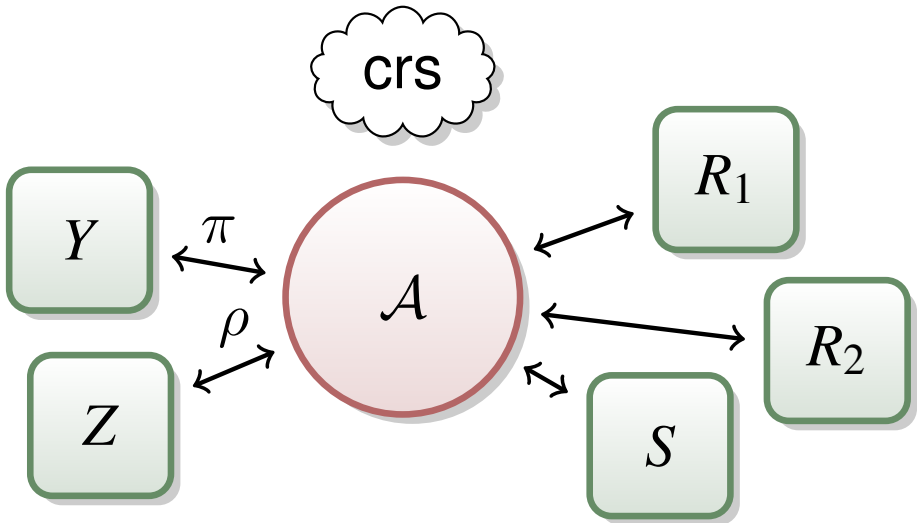
Prior Efficient Protocols

- Two messages: [NP01,AIR01,Tau05,CS02] – *'half simulation'*
- Full simulation: [Lin08] – *blowup, extra rounds*
- 'Adaptive selection:' [CNS07,GH07] – bilinear, many rounds

Will it *COMPOSE* ?

UC framework [Can01]

Requires *setup* [CF01]



COMPOSABILITY aids *EFFICIENCY*

- Stronger OT variants, specific assumptions, 4+ messages [JS07,GMY04,DN03,GH08]

A New OT Framework

Main Attractions

- ✔ *Round-optimal* – two messages

A New OT Framework

Main Attractions

- ✔ *Round-optimal* – two messages
- ✔ *Efficient* – computation & bandwidth

A New OT Framework

Main Attractions

- ✔ *Round-optimal* – two messages
- ✔ *Efficient* – computation & bandwidth
- ✔ *Universally composable* – static corruption, CRS setup

A New OT Framework

Main Attractions

- ✔ *Round-optimal* – two messages
- ✔ *Efficient* – computation & bandwidth
- ✔ *Universally composable* – static corruption, CRS setup
- ✔ *Realizable* – DDH, QR, DCR, *worst-case lattice assumptions*

A New OT Framework

Main Attractions

- ✔ *Round-optimal* – two messages
- ✔ *Efficient* – computation & bandwidth
- ✔ *Universally composable* – static corruption, CRS setup
- ✔ *Realizable* – DDH, QR, DCR, *worst-case lattice assumptions*

Bonus Features

- *Unbounded* CRS reuse (JUC framework [CR03])
- *Statistical* security for either party
- *Simple* & symmetric proof

A New OT Framework

Main Attractions

- ✔ *Round-optimal* – two messages
- ✔ *Efficient* – computation & bandwidth
- ✔ *Universally composable* – static corruption, CRS setup
- ✔ *Realizable* – DDH, QR, DCR, *worst-case lattice assumptions*

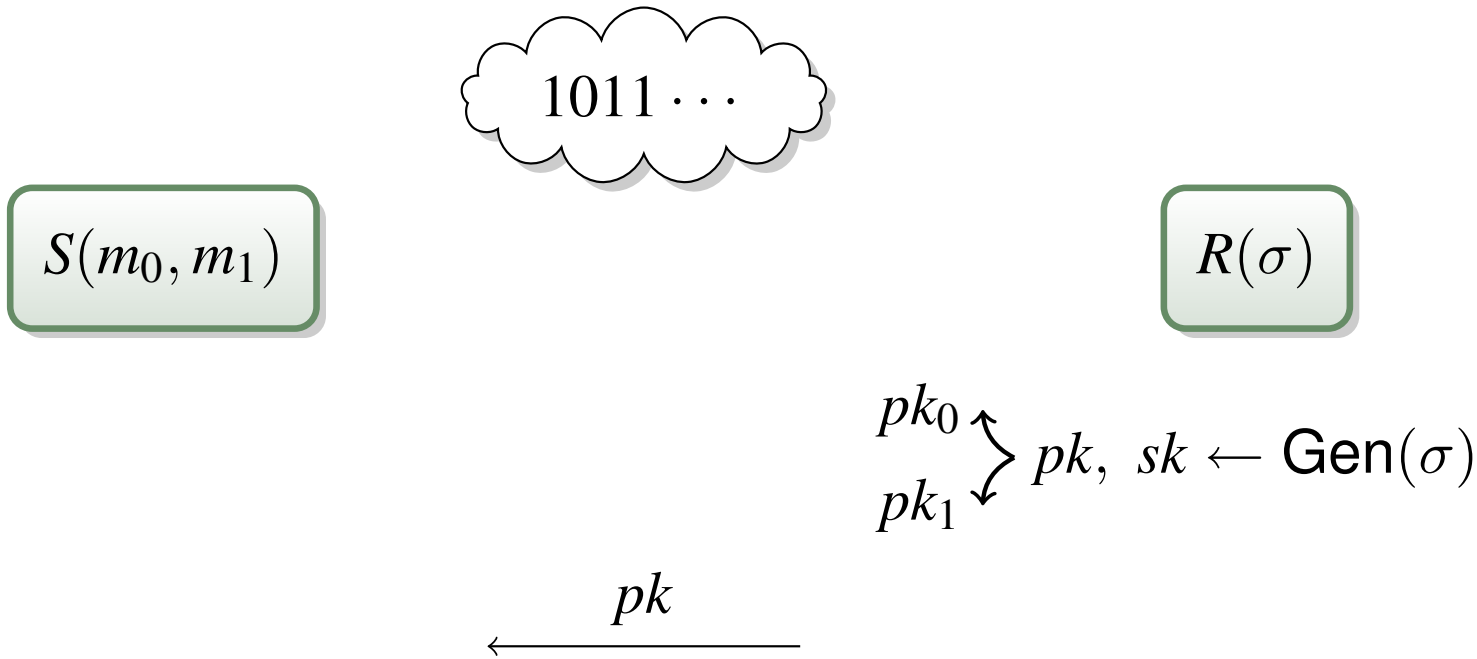
Bonus Features

- *Unbounded* CRS reuse (JUC framework [CR03])
- *Statistical* security for either party
- *Simple* & symmetric proof

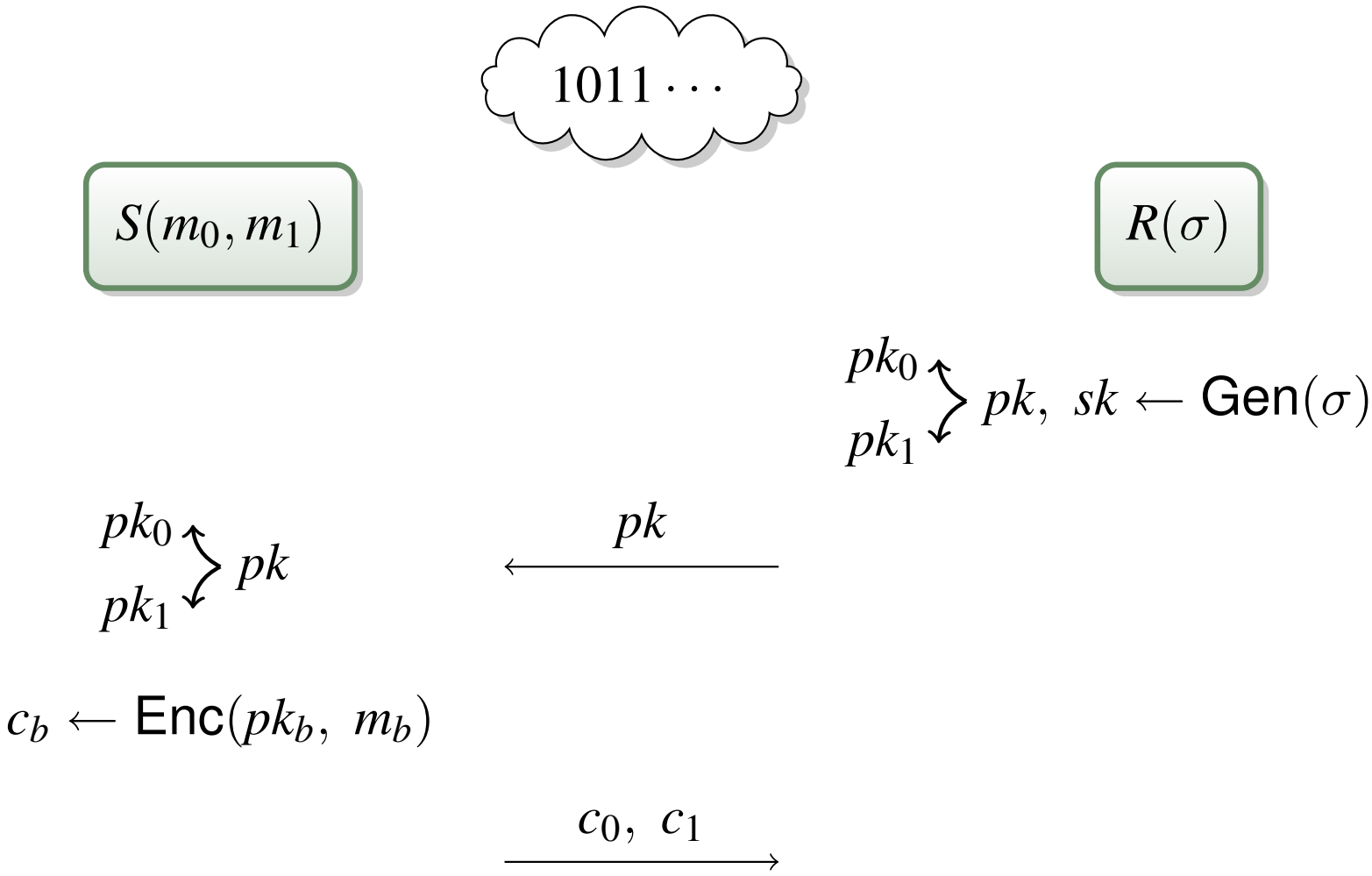
Conceptual Tools

- *Messy* public keys (‘*me*ssage-lo*ssy*’) aka ‘meaningless’ [KN08]
- New abstraction: *Dual-mode* cryptosystem

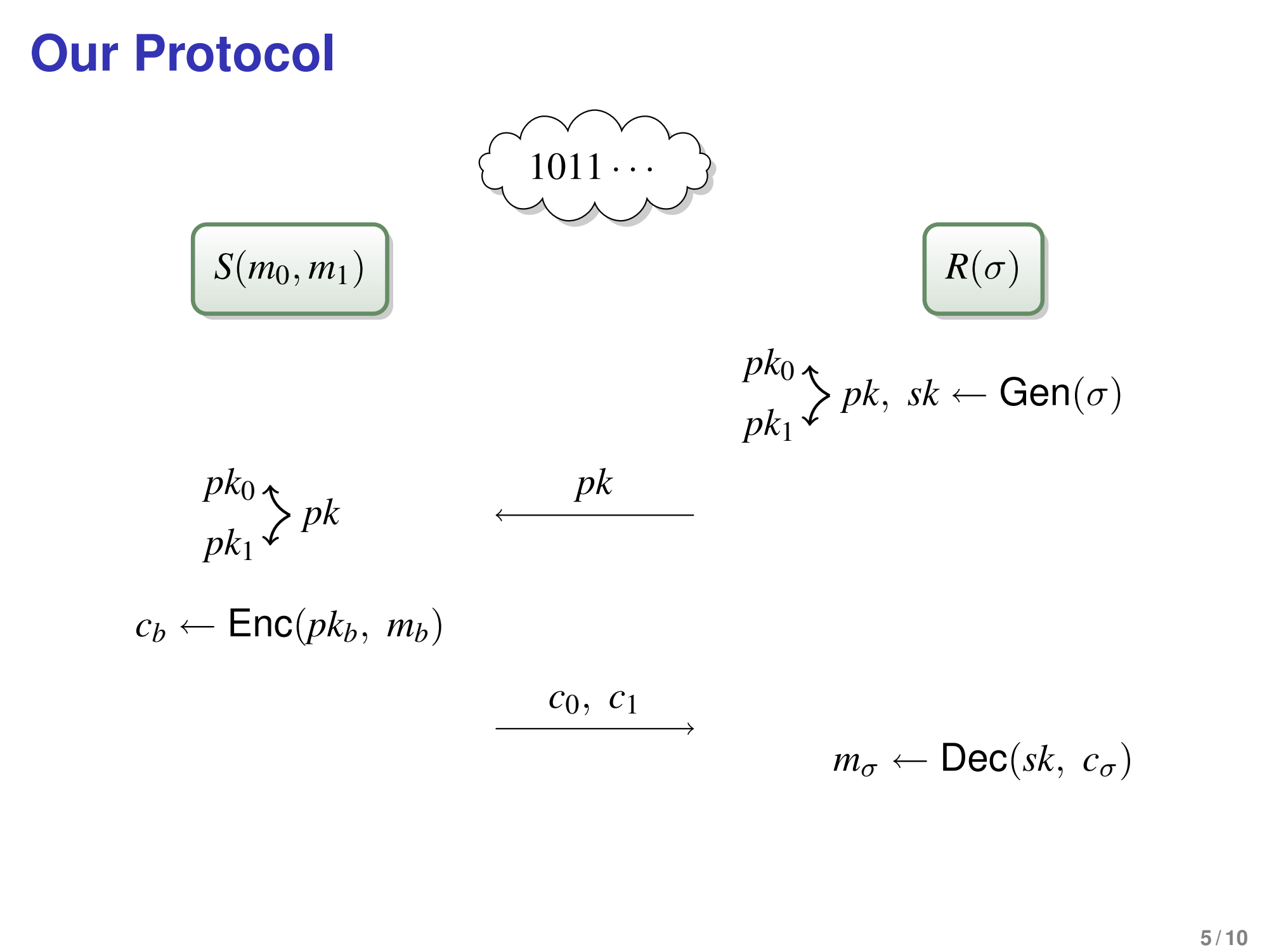
Our Protocol



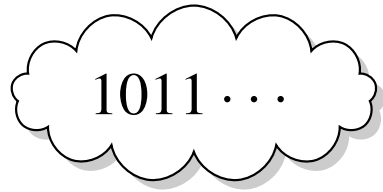
Our Protocol



Our Protocol



Our Protocol



$S(m_0, m_1)$

$R(\sigma)$

pk_0, pk_1 $\leftrightsquigarrow$ $pk,\ sk \leftarrow \mathsf{Gen}(\sigma)$

pk_0, pk_1 $\leftrightsquigarrow$ pk

$\xleftarrow{\quad pk \quad}$

$c_b \leftarrow \mathsf{Enc}(pk_b,\ m_b)$

$\xrightarrow{\quad c_0,\ c_1 \quad}$

$m_\sigma \leftarrow \mathsf{Dec}(sk,\ c_\sigma)$

Needed: *Dual-mode cryptosystem*

Messy Encryption

Decryptable Public Keys

$$\mathsf{Enc}(pk,\ m_0) \quad \overset{c}{\approx} \quad \mathsf{Enc}(pk,\ m_1)$$

- Decrypt with sk.

Messy Encryption

Decryptable Public Keys

$$\mathrm{Enc}(pk,\ m_0) \overset{c}{\approx} \mathrm{Enc}(pk,\ m_1)$$

- Decrypt with sk.

Messy Public Keys

$$\mathrm{Enc}(pk,\ m_0) \overset{s}{\approx} \mathrm{Enc}(pk,\ m_1)$$

- Statistically secure! (Decryption impossible.)

Messy Encryption

Decryptable Public Keys

$$\mathrm{Enc}(pk,\ m_0) \overset{c}{\approx} \mathrm{Enc}(pk,\ m_1)$$

- Decrypt with sk.

Messy Public Keys

$$\mathrm{Enc}(pk,\ m_0) \overset{s}{\approx} \mathrm{Enc}(pk,\ m_1)$$

- Statistically secure! (Decryption impossible.)

Cryptosystems with Messy Keys

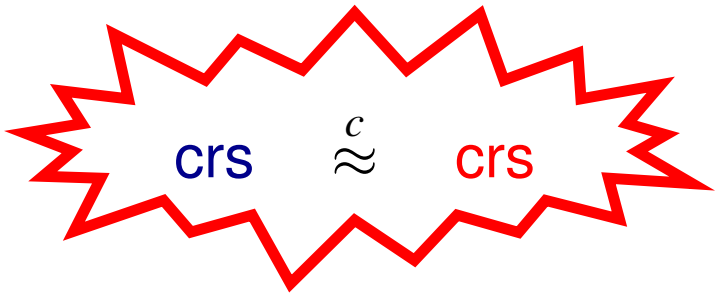
- Cocks ID-based [Coc01]
- Lattice-based [AD97, Reg03, Reg05]
- ElGamal, Paillier variants [ElG84,Pai99]

Dual-Mode Cryptosystem

$$\{\text{dec}, \text{mes}\} \ni \text{mode} \longrightarrow \boxed{\text{Setup}} \longrightarrow (\text{crs},\ \text{trap})$$

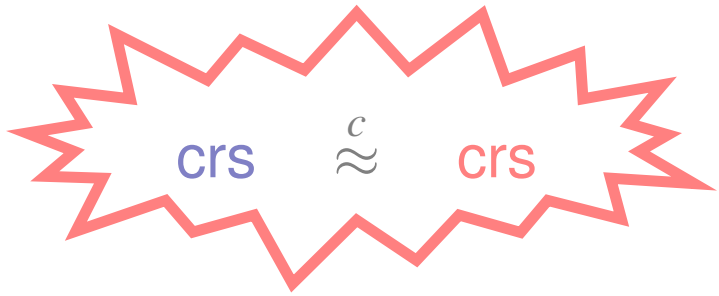
Dual-Mode Cryptosystem

$$\{\mathsf{dec}, \mathsf{mes}\} \ni \mathsf{mode} \longrightarrow \boxed{\mathsf{Setup}} \longrightarrow (\mathsf{crs},\ \mathsf{trap})$$



Dual-Mode Cryptosystem

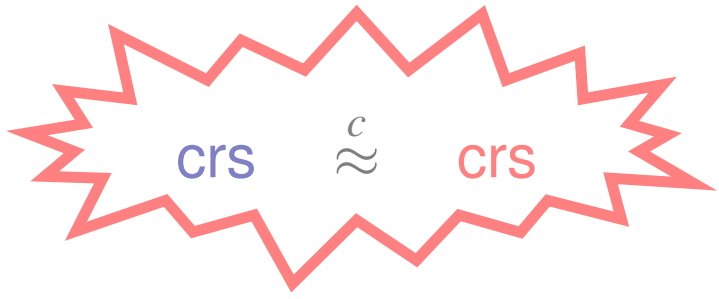
$\{\text{dec}, \text{mes}\} \ni \text{mode} \longrightarrow \boxed{\text{Setup}} \longrightarrow (\text{crs},\ \text{trap})$



$\sigma \longrightarrow \boxed{\text{Gen}} \longrightarrow sk_\sigma,\ pk \begin{cases} \nearrow pk_\sigma \\ \searrow pk_{1-\sigma} \end{cases}$

Dual-Mode Cryptosystem

$\{\text{dec}, \text{mes}\} \ni \text{mode} \longrightarrow$ Setup $\longrightarrow (\text{crs}, \text{trap})$



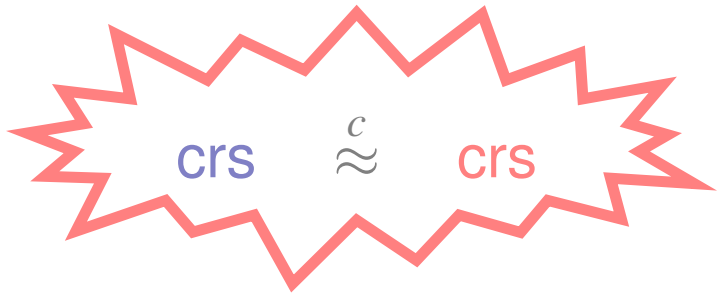
$\sigma \longrightarrow$ Gen $\longrightarrow sk_\sigma, pk \begin{cases} \nearrow pk_\sigma \\ \searrow pk_{1-\sigma} \end{cases}$

$m \longrightarrow$ Enc $\longrightarrow$ Dec $\longrightarrow m$

(pk_σ ↓ Enc, sk_σ ↓ Dec)

Dual-Mode Cryptosystem

$\{\mathsf{dec}, \mathsf{mes}\} \ni \mathsf{mode} \longrightarrow \boxed{\mathsf{Setup}} \longrightarrow (\mathsf{crs},\ \mathsf{trap})$



$\sigma \longrightarrow \boxed{\mathsf{Gen}} \longrightarrow sk_\sigma,\ pk \begin{cases} \nearrow pk_\sigma \\ \searrow pk_{1-\sigma} \end{cases}$

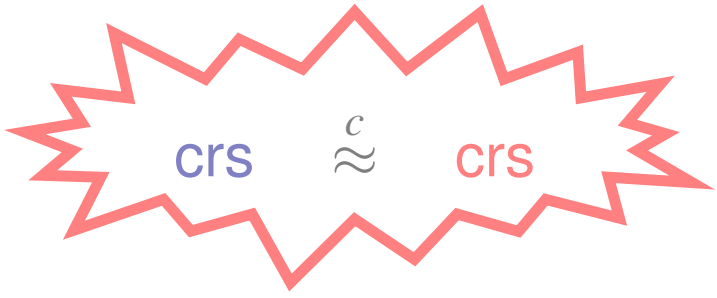
$m \longrightarrow \boxed{\mathsf{Enc}} \longrightarrow \boxed{\mathsf{Dec}} \longrightarrow m$

(pk_σ ↓ Enc; sk_σ ↓ Dec)

$\mathsf{trap} \downarrow \boxed{\mathsf{TrapGen}} \longrightarrow sk_0,\ sk_1,\ pk \begin{cases} \nearrow pk_0 \\ \searrow pk_1 \end{cases}$

Dual-Mode Cryptosystem

$\{\text{dec}, \text{mes}\} \ni \text{mode} \longrightarrow$ Setup $\longrightarrow (\text{crs},\ \text{trap})$



$\sigma \longrightarrow$ Gen $\longrightarrow sk_\sigma,\ pk \begin{cases} pk_\sigma \\ pk_{1-\sigma} \end{cases}$

$m \longrightarrow$ Enc $\longrightarrow$ Dec $\longrightarrow m$

(pk_σ ↓ Enc; sk_σ ↓ Dec)

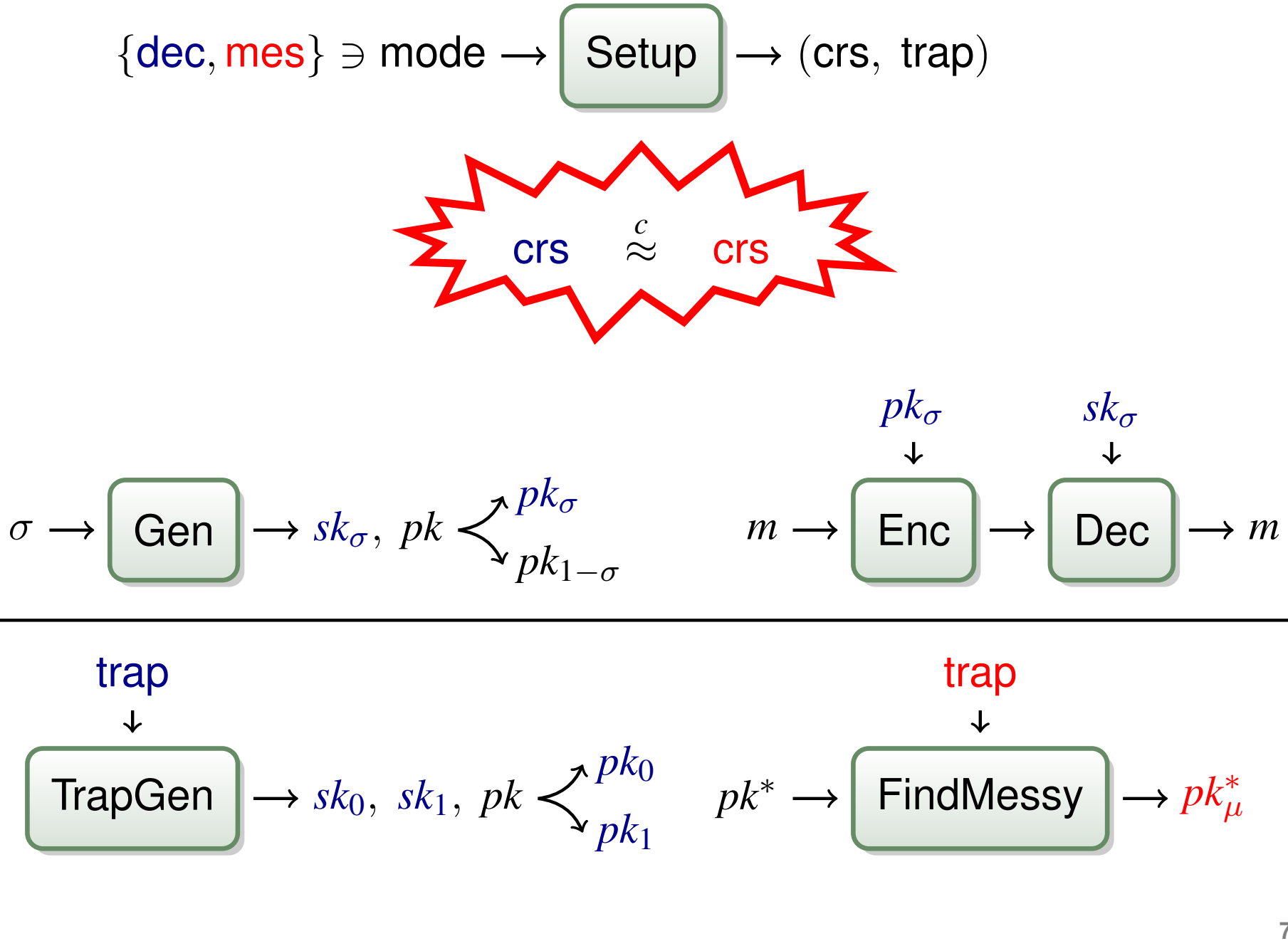
---

trap ↓ TrapGen $\longrightarrow sk_0,\ sk_1,\ pk \begin{cases} pk_0 \\ pk_1 \end{cases}$

$pk^* \longrightarrow$ FindMessy $\longrightarrow pk^*_\mu$

(trap ↓ FindMessy)

Dual-Mode Cryptosystem



Security

Main Theorem

Dual-mode cryptosystem $\Longrightarrow$ 2-message UC-secure OT

Security

Main Theorem

Dual-mode cryptosystem $\Longrightarrow$ 2-message UC-secure OT

Proof Outline

1. $\forall$ real $\mathcal{S}^*$, $\exists$ ideal $\mathcal{S}$ (TrapGen)

$$\text{REAL}(\mathcal{S}^*, \text{ crs}) \overset{s}{\approx} \text{IDEAL}(\mathcal{S})$$

Security

Main Theorem

Dual-mode cryptosystem $\Longrightarrow$ 2-message UC-secure OT

Proof Outline

1. $\forall$ real S^*, $\exists$ ideal $\mathcal{S}$ (TrapGen)

$$\text{REAL}(S^*, \text{crs}) \overset{s}{\approx} \text{IDEAL}(\mathcal{S})$$

2. $\forall$ real R^*, $\exists$ ideal $\mathcal{R}$ (FindMessy)

$$\text{REAL}(R^*, \text{crs}) \overset{s}{\approx} \text{IDEAL}(\mathcal{R})$$

Security

Main Theorem

Dual-mode cryptosystem $\Longrightarrow$ 2-message UC-secure OT

Proof Outline

1. $\forall$ real S^*, $\exists$ ideal $\mathcal{S}$ (TrapGen)

$$\text{REAL}(S^*, \text{crs}) \overset{s}{\approx} \text{IDEAL}(\mathcal{S})$$

2. $\forall$ real R^*, $\exists$ ideal $\mathcal{R}$ (FindMessy)

$$\text{REAL}(R^*, \text{crs}) \overset{s}{\approx} \text{IDEAL}(\mathcal{R})$$

3. $\forall$ real $P^* \in \{R^*, S^*\}$, (Setup)

$$\text{REAL}(P^*, \text{crs}) \overset{c}{\approx} \text{REAL}(P^*, \text{crs})$$

Security

Main Theorem

Dual-mode cryptosystem $\Longrightarrow$ 2-message UC-secure OT

Proof Outline

1. $\forall$ real S^*, $\exists$ ideal $\mathcal{S}$ (TrapGen)

$$\text{REAL}(S^*, \text{crs}) \overset{s}{\approx} \text{IDEAL}(\mathcal{S})$$

2. $\forall$ real R^*, $\exists$ ideal $\mathcal{R}$ (FindMessy)

$$\text{REAL}(R^*, \text{crs}) \overset{s}{\approx} \text{IDEAL}(\mathcal{R})$$

3. $\forall$ real $P^* \in \{R^*, S^*\}$, (Setup)

$$\text{REAL}(P^*, \text{crs}) \overset{c}{\approx} \text{REAL}(P^*, \text{crs})$$

Security in decryption mode (cf. [GOS06]):

✔ $\text{REAL}(R^*, \text{crs}) \overset{c}{\approx} \text{REAL}(R^*, \text{crs}) \overset{s}{\approx} \text{IDEAL}(\mathcal{R})$

Security

Main Theorem

Dual-mode cryptosystem $\Longrightarrow$ 2-message UC-secure OT

Proof Outline

1. $\forall$ real S^*, $\exists$ ideal $\mathcal{S}$ (TrapGen)

$$\text{REAL}(S^*, \text{crs}) \overset{s}{\approx} \text{IDEAL}(\mathcal{S})$$

2. $\forall$ real R^*, $\exists$ ideal $\mathcal{R}$ (FindMessy)

$$\text{REAL}(R^*, \text{crs}) \overset{s}{\approx} \text{IDEAL}(\mathcal{R})$$

3. $\forall$ real $P^* \in \{R^*, S^*\}$, (Setup)

$$\text{REAL}(P^*, \text{crs}) \overset{c}{\approx} \text{REAL}(P^*, \text{crs})$$

Security in messy mode:

✔ $\text{REAL}(S^*, \text{crs}) \overset{c}{\approx} \text{REAL}(S^*, \text{crs}) \overset{s}{\approx} \text{IDEAL}(\mathcal{S})$

Quadratic Residuosity Construction

Cocks Encryption [Coc01]

- Global: $N = pq$
- Decryptable keys: secret $x \in \mathbb{Z}_N$, public $y = x^2 \in \mathbb{QR}_N$.
- Messy public key: $y \notin \mathbb{QR}_N$

Quadratic Residuosity Construction

Cocks Encryption [Coc01]

- Global: $N = pq$
- Decryptable keys: secret $x \in \mathbb{Z}_N$, public $y = x^2 \in \mathbb{QR}_N$.
- Messy public key: $y \notin \mathbb{QR}_N$

Our Construction

Decryption mode

$\text{crs} = (N,\ z \in \mathbb{QR}_N)$

$\text{trap} = \sqrt{z}$

Messy mode

$\text{crs} = (N,\ z \in \mathbb{J}_N \backslash \mathbb{QR}_N)$

$\text{trap} = (p, q)$

Quadratic Residuosity Construction

Cocks Encryption [Coc01]

- Global: $N = pq$
- Decryptable keys: secret $x \in \mathbb{Z}_N$, public $y = x^2 \in \mathbb{QR}_N$.
- Messy public key: $y \notin \mathbb{QR}_N$

Our Construction

Decryption mode	**Messy mode**
$\text{crs} = (N,\ z \in \mathbb{QR}_N)$	$\text{crs} = (N,\ z \in \mathbb{J}_N \backslash \mathbb{QR}_N)$
$\text{trap} = \sqrt{z}$	$\text{trap} = (p, q)$

$$\mathbb{Z}_N \ni pk = y \begin{cases} y_0 = y \cdot z^0 \\ y_1 = y \cdot z^1 \end{cases}$$

Quadratic Residuosity Construction

Cocks Encryption [Coc01]

- Global: $N = pq$
- Decryptable keys: secret $x \in \mathbb{Z}_N$, public $y = x^2 \in \mathbb{QR}_N$.
- Messy public key: $y \notin \mathbb{QR}_N$

Our Construction

Decryption mode	**Messy mode**
$\text{crs} = (N,\ z \in \mathbb{QR}_N)$	$\text{crs} = (N,\ z \in \mathbb{J}_N \backslash \mathbb{QR}_N)$
$\text{trap} = \sqrt{z}$	$\text{trap} = (p, q)$

$$\mathbb{Z}_N \ni pk = y \begin{cases} y_0 = y \cdot z^0 \\ y_1 = y \cdot z^1 \end{cases}$$

TrapGen:	FindMessy:
$sk_0 = \sqrt{y},\ sk_1 = \sqrt{y} \cdot \sqrt{z}$	$y \notin \mathbb{QR}_N \quad \text{or} \quad y \cdot z \notin \mathbb{QR}_N$

Some Open Problems

1. Adaptive corruptions? (Progress: [GWZ])

Some Open Problems

1. Adaptive corruptions? (Progress: [GWZ])
2. Alternate setup? (GUC framework? [CDPW07])

Some Open Problems

1. Adaptive corruptions? (Progress: [GWZ])
2. Alternate setup? (GUC framework? [CDPW07])
3. String OT from QR, lattices? (Note: [BGH07] isn't messy!)

Some Open Problems

1. Adaptive corruptions? (Progress: [GWZ])
2. Alternate setup? (GUC framework? [CDPW07])
3. String OT from QR, lattices? (Note: [BGH07] isn't messy!)